UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/709,128	04/15/2004	Peter A. Franaszek	YOR920040079US1	3127

23413 7590 05/04/2006

CANTOR COLBURN, LLP
55 GRIFFIN ROAD SOUTH
BLOOMFIELD, CT 06002

EXAMINER	
IWASHKO, LEV	

ART UNIT	PAPER NUMBER
2186	

DATE MAILED: 05/04/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/709,128	FRANASZEK ET AL.	
	Examiner	Art Unit	
	Lev I. Iwashko	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 April 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-29* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-29* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *15 April 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date ~~4/14/04~~, *4/29/04*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following are quotations of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1-7, 10, 14-18, and 21 are rejected under U.S.C. 102(b) as being anticipated by Keltcher et al. (US Patent 6,314,494 B1).

Claim 1. A system for memory management, the system comprising:

- a tag controlled buffer in communication with a memory device, said memory device including a plurality of pages divided into a plurality of individually addressable lines, wherein said tag controlled buffer includes: *(Column 1, lines 29-33 – State the following: "Instructions and data received from the main memory by the controller for execution are also stored in the high speed cache memory. Therefore, the controller has ready access to the most recently executed instructions and data if the same instructions or data be needed again by the controller." Column 1, lines 38-42 – State the following: "It is important to keep track of which lines of code and data are stored in the cache memory. One technique is to use TAG cache memory which includes memory locations for storing TAG addresses that correspond to addresses of the particular information stored in the cache memory." (Column 4, lines 17-27 – State the following: "FIG. 3 shows the address bits of the Controller BUS 25 of FIG. 1. The address bits of the controller address bus 25 include column address bits 42, line address bits 44 and TAG address bits 46. This embodiment includes five column address bits (bits 4:0), seven line address bits (bits 11:5),*

and twenty TAG address bits (bits 31:12). Five column address bits are required because this embodiment of the size configurable data buffer 12 includes 32 columns. Seven line address bits are required because this embodiment of the size configurable data buffer 12 includes 128 lines (rows) of memory")

- a prefetch buffer including at least one of the individually addressable lines from the memory device; *(Column 3, lines 11-32 – State the following: "As shown in the drawings for purposes of illustration, the invention is embodied in a size configurable data buffer. The size configurable data buffer includes two caches: data and prefetch. The size of the prefetch cache is adjustable. The size configurable data buffer can be implemented with a single SRAM (Static Random Access Memory) circuit. FIG. 1 shows an embodiment of the invention. This embodiment includes a size configurable data buffer 12. The size configurable data buffer 12 includes data cache and prefetch cache. Mask circuitry 14 determines the allocation between data cache and prefetch cache within the size configurable data buffer 12. The embodiment of FIG. 1 shows the data cache and prefetch cache within a single SRAM circuit, the size configurable data buffer 12. However, the data cache and prefetch cache can be divided into multiple SRAM circuits. A single SRAM circuit or integrated circuit is generally less expensive than two circuits or integrated circuits. The embodiment shown in FIG. 1 also includes an address recovery SRAM 16, a TAG SRAM 18, a TAG comparator 20, a controller (CPU) 22, a controller address bus (Controller Bus [31:0] 25), a main memory address bus (Main Memory BUS [31:0] 27) and a main memory 24")*
- and a tag cache in communication with the prefetch buffer, the tag cache including a plurality of tags, wherein each said tag is associated with one of the pages in the memory device, each said tag includes a pointer to at least one of the lines in the prefetch buffer, and access to

the lines in the prefetch buffer is controlled by the tag cache. *(Column 6, lines 4-25 – State the following: "The store mode occurs when the controller 22 is writing new data to the size configurable data buffer 12 (data cache or prefetch cache). Initially, the CASTOUT input is deactivated (set false). The PREFETCH input is activated (set true) if the controller 22 recognizes the data as prefetch data. The controller 22 generates the address bits of the controller address bus 25 that correspond to the address of the location the data is to be written. When the address bits of the location match the line address bits stored within the address recovery SRAM 16 and the TAG address bits stored within the TAG SRAM 18, a TAG hit occurs and the controller 22 writes the data to the size configurable data buffer 12. When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data. The line address bits and the TAG address bits of the new data are stored in the address recovery SRAM 16 and the TAG SRAM 18 for future reference")*

Claim 2. The system of claim 1 wherein access includes at least one of inserting new lines into the prefetch buffer, deleting one of the lines from the prefetch buffer and reading one of the lines in the prefetch buffer. *(Column 5,lines 38-40 – State the following: "here are generally three major modes of operation. The three modes include a castout mode, a load mode and a store mode")*

Claim 3. The system of claim 1 wherein the tag controlled buffer transmits one of the lines in the prefetch buffer to a cache device in response to a command

from the tag cache. *(Column 5, lines 23-30 – State the following: "When f the requested address bits and the stored address bits are identical , a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12. The controller can subsequently utilize the data from the size configurable data buffer 12 that was accessed at the same time as the TAG access")*

Claim 4. The system of claim 1 wherein the prefetch buffer is implemented by a random access memory. *(Column 23, lines 29 – Discloses a TAG SRAM 18)*

Claim 5. The system of claim 1 wherein all of the lines in the prefetch buffer corresponding to one of the tags in the tag cache are deleted in response to the tag being deleted from the tag cache. *(Column 5, lines 41-55 – State the following: "The castout mode occurs when transferring data from the size configurable data buffer 12 (data cache or prefetch cache) to main memory 24. In the castout mode, the CASTOUT input is activated (set true) by the controller 22. Therefore, the mask circuitry 14 cannot modify the line address bits. The controller 22 accesses the data within the size configurable data buffer 12. The address bits of the main memory address bus 27 are constructed from the columns bits of the controller address bus 25, and the data output of the address recovery SRAM 16 and the data output of the TAG SRAM 18. A load/store control line from the controller 22 drives read/write (RWB) inputs to the size configurable data buffer 12, the address recovery SRAM 16 and the TAG SRAM 18 to a read mode. The data transfer from the size configurable data buffer 12 to main memory 24 is then completed")*

Claim 6. The system of claim 1 wherein the system further comprises instructions to implement a replacement algorithm for the tags in the tag cache and for the lines in the prefetch buffer, wherein upon insertion of a new tag into the tag cache, a sufficient number of the tags are removed to make space

for the new tag and for the prefetch lines associated with the new tag. *(Column 6, lines 4-25 – State the following: "The store mode occurs when the controller 22 is writing new data to the size configurable data buffer 12 (data cache or prefetch cache). Initially, the CASTOUT input is deactivated (set false). The PREFETCH input is activated (set true) if the controller 22 recognizes the data as prefetch data. The controller 22 generates the address bits of the controller address bus 25 that correspond to the address of the location the data is to be written. When the address bits of the location match the line address bits stored within the address recovery SRAM 16 and the TAG address bits stored within the TAG SRAM 18, a TAG hit occurs and the controller 22 writes the data to the size configurable data buffer 12. When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data. The line address bits and the TAG address bits of the new data are stored in the address recovery SRAM 16 and the TAG SRAM 18 for future reference")*

Claim 7. The system of claim 6 wherein the tags that are removed are written back to a lower level memory. *(Column 6, lines 15-24 – State the following: "When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data")*

Claim 10. The system of claim 1 wherein the system further comprises instructions to implement a method comprising updating the tags in the tag cache in response to the lines in the prefetch buffer being inserted, deleted, modified and referenced. *(Columns 5 and 6 – Clearly show that the tags are updated upon insertion, deletion, modification, and reference)*

Claim 14. A system for memory management, the system comprising:

- a random access memory *(Column 23, lines 29 – Discloses a TAG SRAM 18)*
- including at least one line, wherein each line is associated with a page in a memory device and space in the random access memory is allocated on per line basis; *(Column 1, lines 29-33 – State the following: "Instructions and data received from the main memory by the controller for execution are also stored in the high speed cache memory. Therefore, the controller has ready access to the most recently executed instructions and data if the same instructions or data be needed again by the controller." Column 1, lines 38-42 – State the following: "It is important to keep track of which lines of code and data are stored in the cache memory. One technique is to use TAG cache memory which includes memory locations for storing TAG addresses that correspond to addresses of the particular information stored in the cache memory." (Column 4, lines 17-27 – State the following: "FIG. 3 shows the address bits of the Controller BUS 25 of FIG. 1. The address bits of the controller address bus 25 include column address bits 42, line address bits 44 and TAG address bits 46. This embodiment includes five column address bits (bits 4:0), seven line address bits (bits 11:5), and twenty TAG address bits (bits 31:12). Five column address bits are required because this embodiment of the size configurable data buffer 12 includes 32 columns. Seven line address bits are required because this embodiment of the size configurable data buffer 12 includes 128 lines (rows) of memory")*

- and a first cache device including a plurality of tags, wherein each tag corresponds to one of the pages in the memory device and each tag indicates the location in the random access memory of the at least one line associated with the page. *(Column 6, lines 4-25 – State the following: "The store mode occurs when the controller 22 is writing new data to the size configurable data buffer 12 (data cache or prefetch cache). Initially, the CASTOUT input is deactivated (set false). The PREFETCH input is activated (set true) if the controller 22 recognizes the data as prefetch data. The controller 22 generates the address bits of the controller address bus 25 that correspond to the address of the location the data is to be written. When the address bits of the location match the line address bits stored within the address recovery SRAM 16 and the TAG address bits stored within the TAG SRAM 18, a TAG hit occurs and the controller 22 writes the data to the size configurable data buffer 12. When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data. The line address bits and the TAG address bits of the new data are stored in the address recovery SRAM 16 and the TAG SRAM 18 for future reference")*

Claim 15. The system of claim 14 further comprising a computer processor, the computer processor including instructions *(Column 1, lines 13-20 – State the following: "Most modern computer systems include a controller and a main memory. The speed at which the controller can decode and execute instructions to process data has for some time exceeded the speed at which instructions and data can be transferred from main memory to the*

controller. In an attempt to reduce the problems cause by this mismatch, most computer systems include a cache memory between the controller and main memory")

- to transmit one of the lines in the random access memory to a second cache device in response to a command from the first cache device. *(Column 5, lines 23-27 – State the following: "When the requested address bits and the stored address bits are identical , a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12")*

Claim 16. The system of claim 14 wherein all of the lines in the random access memory corresponding to one of the tags in the first cache device are deleted in response to the tag being deleted from the tag cache. *(Column 5, lines 41-55 – State the following: "The castout mode occurs when transferring data from the size configurable data buffer 12 (data cache or prefetch cache) to main memory 24. In the castout mode, the CASTOUT input is activated (set true) by the controller 22. Therefore, the mask circuitry 14 cannot modify the line address bits. The controller 22 accesses the data within the size configurable data buffer 12. The address bits of the main memory address bus 27 are constructed from the columns bits of the controller address bus 25, and the data output of the address recovery SRAM 16 and the data output of the TAG SRAM 18. A load/store control line from the controller 22 drives read/write (RWB) inputs to the size configurable data buffer 12, the address recovery SRAM 16 and the TAG SRAM 18 to a read mode. The data transfer from the size configurable data buffer 12 to main memory 24 is then completed")*

Claim 17. The system of claim 14 further comprising a computer processor, the computer processor including instructions to implement a replacement algorithm for the tags in the first cache device and for the lines in the random access memory, wherein upon insertion of a new tag into the first

cache device, a sufficient number of the tags are removed to make space for the new tag and for the prefetch lines associated with the new tag. *(Column 6, lines 4-25 – State the following: "The store mode occurs when the controller 22 is writing new data to the size configurable data buffer 12 (data cache or prefetch cache). Initially, the CASTOUT input is deactivated (set false). The PREFETCH input is activated (set true) if the controller 22 recognizes the data as prefetch data. The controller 22 generates the address bits of the controller address bus 25 that correspond to the address of the location the data is to be written. When the address bits of the location match the line address bits stored within the address recovery SRAM 16 and the TAG address bits stored within the TAG SRAM 18, a TAG hit occurs and the controller 22 writes the data to the size configurable data buffer 12. When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data. The line address bits and the TAG address bits of the new data are stored in the address recovery SRAM 16 and the TAG SRAM 18 for future reference")*

Claim 18. The system of claim 17 wherein the tags that are removed are written back to a lower level memory. *(Column 6, lines 15-24 – State the following: "When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size*

configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data")

Claim 21. The system of claim 14 further comprising a computer processor, the computer processor including instructions to implement a method for updating the tags in the first cache device in response to the lines in the random access memory being inserted, deleted, modified and referenced. *(Columns 5 and 6 – Clearly show that the tags are updated upon insertion, deletion, modification, and reference)*

3. Claim 11 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher as applies to Claim 1, further in view of Dean (US Patent 5,544,342).

Keltcher teaches the limitations of claim 11 as follows:

- accessing the tag cache to determine if one of the lines in the prefetch buffer corresponds to the fault page identifier and the fault line identifier; *(Column 5, lines 18-31 – State the following: "The TAG comparator 20 compares address bits of the controller address bus 25 (line address and TAG address bits) of data requested by the controller 22 with the address bits of the main memory address bus 27 (line address and TAG address bits) stored in the corresponding address recovery SRAM 16 and TAG SRAM 18 line address or modified line address locations. When f the requested address bits and the stored address bits are identical, a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12. The controller can subsequently utilize the data from the size configurable data buffer 12 that was accessed at the same time as the TAG access. If a TAG miss occurs, the controller must access the requested data from the main memory 24")*

Keltcher's invention differs from the claimed invention in that there is no specific reference to faults.

Keltcher fails to teach the entirety of Claim 11. However, Dean teaches the following aspects of Claim 11:

> The system of claim 1 wherein the system further includes instructions to implement a method comprising:
>
> - receiving a fault notification from a cache device, the fault notification including a fault page identifier and a fault line identifier; *(Column 31, lines 57-60 – State the following: "For fetch-on-fault, a fetch to the next level in the memory hierarchy is initiated in response to a cache miss and retrieves more than one cache block of data (fetch size >block size)")*
> - transmitting the line corresponding to the fault page identifier and the fault line identifier from the prefetch buffer to the cache device in response to locating the line corresponding to the fault page identifier and the fault line identifier in the prefetch buffer; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*
> - and updating the tag cache to reflect the transmitting. *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*

It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" before him at the time the invention was made, to combine the inventions to utilize faults in order to make the system more efficient and diminish error handling problems.

4. Claim 24 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher, further in view of Dean (US Patent 5,544,342).

Keltcher teaches the limitations of claim 24 as follows:

> - accessing a second cache device to determine if a line in a random access memory corresponds to the fault page identifier and the fault line identifier, wherein: the random access memory includes at least

one line associated with a page in a memory device; *(Column 5, lines 18-31 – State the following: "The TAG comparator 20 compares address bits of the controller address bus 25 (line address and TAG address bits) of data requested by the controller 22 with the address bits of the main memory address bus 27 (line address and TAG address bits) stored in the corresponding address recovery SRAM 16 and TAG SRAM 18 line address or modified line address locations. When f the requested address bits and the stored address bits are identical, a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12. The controller can subsequently utilize the data from the size configurable data buffer 12 that was accessed at the same time as the TAG access. If a TAG miss occurs, the controller must access the requested data from the main memory 24")*

- the second cache device includes a plurality of tags each corresponding to one of the pages in the memory device; *(Column 5, lines 23-27 – State the following: "When the requested address bits and the stored address bits are identical , a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12")*
- and each tag indicates the location in the random access memory of the at least one line associated with the page corresponding to the tag; *(Column 6, lines 4-25 – State the following: "The store mode occurs when the controller 22 is writing new data to the size configurable data buffer 12 (data cache or prefetch cache). Initially, the CASTOUT input is deactivated (set false). The PREFETCH input is activated (set true) if the controller 22 recognizes the data as prefetch data. The controller 22 generates the address bits of the controller address bus 25 that correspond to the address of the location the data is to be written. When the address bits of the location match the line address bits stored within the address recovery SRAM 16 and the TAG address bits stored within the TAG SRAM 18, a TAG hit occurs and the controller 22 writes the data to the size configurable data buffer 12. When a TAG miss occurs and the data located in the size configurable data buffer 12 at the location the new data is to be written has previously been modified, then the controller 22 activates the CASTOUT input and transfers that data to the main memory 24. The controller then deactivates the CASTOUT line and writes the new data to the size configurable data buffer 12. Again, the PREFETCH input is activated when the controller 22 recognizes the data as prefetch data. The line address bits and the TAG address bits of the new data*

are stored in the address recovery SRAM 16 and the TAG SRAM 18 for future reference")

Keltcher's invention differs from the claimed invention in that there is no specific reference to faults.

Keltcher fails to teach the entirety of Claim 24. However, Dean teaches the following aspects of Claim 24:

A method for memory management, the method comprising:

- receiving a fault notification from a first cache device, the fault notification including a fault page identifier and a fault line identifier; *(Column 31, lines 57-60 – State the following: "For fetch-on-fault, a fetch to the next level in the memory hierarchy is initiated in response to a cache miss and retrieves more than one cache block of data (fetch size >block size)")*
- transmitting the line corresponding to the fault page identifier and the fault line identifier from the random access memory to the first cache device in response to the accessing resulting in locating the line corresponding to the fault page identifier and the fault line identifier in the random access memory; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*
- and updating the tag in the second cache device corresponding to the fault page identifier to reflect the transmitting. *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*

It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" before him at the time the invention was made, to combine the inventions to utilize faults in order to make the system more efficient and diminish error handling problems.

locating the tag corresponding to the fault page identifier: transmitting a line corresponding to the fault line identifier to the requestor; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*

- and updating the tag corresponding to the fault page identifier to reflect the transmitting; *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*
- and in response to not locating the tag corresponding to the fault page identifier: inserting a new tag corresponding to the fault page identifier into the tag cache; transmitting the line corresponding to the fault line identifier to the requestor; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*
- and inserting the prefetch lines included in the new tag into the prefetch buffer, wherein the inserting is performed via the tag cache. *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*

It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" before him at the time the invention was made, to combine the inventions to utilize faults in order to make the system more efficient and diminish error handling problems.

6. Claim 26 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher as applies to Claim 25, further in view of Dean (US Patent 5,544,342).

Keltcher and Dean teach the limitations of claim 25 for the reasons above.

Keltcher teaches claim 26, which reads as follows: "The method of claim 25 further comprising retrieving the line corresponding to the fault line identifier from a memory device."

5. Claim 25 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher, further in view of Dean (US Patent 5,544,342).

Keltcher teaches the limitations of claim 25 as follows:

- accessing a second cache device to determine if a line in a random access memory corresponds to the fault page identifier and the fault line identifier, wherein: the random access memory includes at least one line associated with a page in a memory device; *(Column 5, lines 18-31 – State the following: "The TAG comparator 20 compares address bits of the controller address bus 25 (line address and TAG address bits) of data requested by the controller 22 with the address bits of the main memory address bus 27 (line address and TAG address bits) stored in the corresponding address recovery SRAM 16 and TAG SRAM 18 line address or modified line address locations. When f the requested address bits and the stored address bits are identical, a TAG hit occurs that indicates to the controller 20 that the requested data is stored in the data cache or the prefetch cache of the size configurable data buffer 12. The controller can subsequently utilize the data from the size configurable data buffer 12 that was accessed at the same time as the TAG access. If a TAG miss occurs, the controller must access the requested data from the main memory 24")*

Keltcher's invention differs from the claimed invention in that there is no specific reference to faults.

Keltcher fails to teach the entirety of Claim 25. However, Dean teaches the following aspects of Claim 25:

A method for memory management, the method comprising:

- receiving a fault notification from a requestor, the fault notification including a fault page identifier and a fault line identifier; *(Column 31, lines 57-60 – State the following: "For fetch-on-fault, a fetch to the next level in the memory hierarchy is initiated in response to a cache miss and retrieves more than one cache block of data (fetch size >block size)")*
- determining if a tag corresponding to the fault page identifier is located in a tag cache, wherein the tag cache includes a plurality of tags, each tag includes at least one pointer to lines in a prefetch buffer, and the tag cache designates at least one prefetch line; in response to

Keltcher states: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor" (Column 32, lines 1-4). It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" to combine the inventions to retrieve the line corresponding to the fault line identifier so that the system would run more accurately.

7. Claim 27 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher as applies to Claim 25, further in view of Dean (US Patent 5,544,342).

Keltcher and Dean teach the limitations of claim 25 for the reasons above.

Keltcher teaches claim 27, which reads as follows: "The method of claim 25 further comprising retrieving the line corresponding to the fault line identifier from the prefetch buffer, wherein the retrieving is via the tag cache." Keltcher states: "Data retrieved from the main memory 24 by the controller 22 for execution can be stored in data cache" (Column 3, lines 30-33). "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor" (Column 32, lines 1-4). It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" to combine the inventions to retrieve the line corresponding to the fault line identifier so that the system would run more accurately.

8. Claim 28 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher as applies to Claim 25, further in view of Dean (US Patent 5,544,342).

Keltcher and Dean teach the limitations of claim 25 for the reasons above.

Keltcher teaches claim 28, which reads as follows: "The method of claim 25 wherein the requestor is a cache device." Keltcher states: "A sixth embodiment of the invention is similar to the fifth embodiment. The sixth embodiment further includes a TAG compare circuit for comparing requested line address bits with line address bits stored in the address recovery SRAM and the TAG SRAM" (Column 2, lines 58-62). It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" to combine the inventions so that the requestor is a cache device so that the system would respond to the direct requests by the cache, making the system more efficient.

9. Claim 29 is rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher, further in view of Dean (US Patent 5,544,342).

Keltcher's invention differs from the claimed invention in that there is no specific reference to faults.

Keltcher fails to teach the entirety of Claim 29. However, Dean teaches the following aspects of Claim 29:

> A computer program product for cache memory management, the computer program product comprising: a storage medium readable by a processing circuit and storing instructions for execution by the processing circuit for performing a method comprising:
>
> - receiving a fault notification from a requester, the fault notification including a fault page identifier and a fault line identifier; *(Column 31,*

lines 57-60 – State the following: "For fetch-on-fault, a fetch to the next level in the memory hierarchy is initiated in response to a cache miss and retrieves more than one cache block of data (fetch size >block size)")

- determining if a tag corresponding to the fault page identifier is located in a tag cache, wherein the tag cache includes a plurality of tags, each tag includes at least one pointer to lines in a prefetch buffer, and the tag cache designates at least one prefetch line; in response to locating the tag corresponding to the fault page identifier: transmitting a line corresponding to the fault line identifier to the requestor; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*
- and updating the tag corresponding to the fault page identifier to reflect the transmitting; *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*
- and in response to not locating the tag corresponding to the fault page identifier: inserting a new tag corresponding to the fault page identifier into the tag cache; transmitting the line corresponding to the fault line identifier to the requestor; *(Column 32, lines 1-4 – State the following: "For tagged-sequential prefetching, a fetch to the next level in the memory hierarchy is initiated in response to a processor request to a cache block which has been fetched but not accessed by the processor")*
- and inserting the prefetch lines included in the new tag into the prefetch buffer, wherein the inserting is performed via the tag cache. *(Column 31, lines 60-62 – State the following: "The additional blocks or subblocks of fetched data are sequentially addressed from the cache miss block address")*

It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher and Dean's "System and Method For Prefetching Information in a Processing System" before him at the time the invention was made, to combine the inventions to utilize faults in order to make the system more efficient and diminish error handling problems.

10. Claims 8-9, 12-13, 19-20, and 22-23 are rejected under 35 U.S.C.103(a) as being unpatentable over Keltcher as applies to Claims 1 and 14, further in view of Teng (US Patent 4,658,351) and Blaner et al. (US Patent 5,423,011).

Keltcher teaches the limitations of claims 1 and 14 for the reasons above.

Keltcher's invention differs from the proposed invention because it fails to reference a page identifier field, a presence bits field, a history vector field, a pointers field and a flags field.

Keltcher fails to teach the following claims:

Claim 8. The system of claim 6 wherein the tags that are removed are written back to a lower level memory and the tags in the lower level memory include a page identifier field and a reference history field.

Claim 9. The system of claim 8 wherein the tags in the lower level memory further include a flags field.

Claim 12. The system of claim 1 wherein each tag in the tag cache includes a page identifier field, a presence bits field, a history vector field, and a pointers field.

Claim 13. The system of claim 12 wherein each tag in the tag cache further includes a flags field.

Claim 19. The system of claim 17 wherein the tags that are removed are written back to a lower level memory and the tags in the lower level memory include a page identifier field and a reference history field.

Claim 20. The system of claim 19 wherein the tags in the lower level memory further include a flags field.

Claim 22. The system of claim 14 wherein each tag in the first cache device includes a page identifier field, a presence bits field, a history vector field, and a pointers field.

Claim 23. The system of claim 22 wherein each tag in the first cache device further includes a flags field.

However, Teng makes the following references:

- "Field 906 (Current Page Number) contains an identifier of the Document Page" (Column 13, lines 29-30)
- "Flag Field 1150" (Column 17, line 39)
- "Each TCB contains a pointer field" (Column 19, line 14)
- "A menu page may be subdivided into multiple areas. Each area is independent of other areas. A line in an area may consist of multiple fields. A field may consist of multiple strings. The first string in the menu field has the "first attributed string" bit set in the string header.

> The last string in a menu field has the "last attributed string" bit set in the string header." (Column 77, lines 64-68)

Furthermore, Blaner states: "The combination of claim 6 wherein said associative storage means further includes a plurality of entries, each entry being comprised of a branch history vector field for storing said branch prediction information, a tag field for associating said branch history vector with said data, and validity field, for indicating the validity of said entry" (Column 8, lines 23-30). It would have been obvious to one of ordinary skill in the art, having the teachings of the "Dynamically Size Configurable Data Buffer for Data Cache and Prefetch Cache Memory" of Keltcher, Teng's "Task control means for a multi-tasking data processing system" and Blaner's "Apparatus for initializing branch prediction information" to combine the inventions to include all of the aforementioned fields to allow for better system organization, thereby increasing efficiency of the system.

Conclusion

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lev I. Iwashko whose telephone number is (571)272-1658. The examiner can normally be reached on M-F (alternating Fridays), from 8-4PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt Kim can be reached on (571)272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Lev Iwashko

PIERRE BATAILLE
PRIMARY EXAMINER
4|30|06